

Sheet 1
Solution

2015

ENG. HAMADA RIZK

Week 2

1. Are each of the following declarations valid or invalid? If any are invalid, why?

	Question	Answer
A)	<code>int ivar; int *iptr = &ivar;</code>	valid
B)	<code>int ivar, *iptr = &ivar;</code>	valid
C)	<code>float fvar; int *iptr = &fvar;</code>	Invalid the pointer of a variable must be of the same type as the variable.
D)	<code>int nums[50], *iptr = nums;</code>	Valid

2. What is the output of the following program assuming array is an array of ints:

Question	Answer
<code>if (&array[] < &array[]) cout << "True"; else cout << "False";</code>	False

3. Assume pint is a pointer variable. Are each of the following statements valid or invalid? If any are invalid, why?

	Question	Answer
A)	<code>pint++;</code>	Valid
B)	<code>pint += x; // Assume x is an int</code>	Valid
C)	<code>pint /= 2 ;</code>	Invalid
D)	<code>pint *= 4 ;</code>	Invalid

4. What is wrong in the following program:

Question	Answer
<code>int numbers[5] = {10, 20, 30, 40, 50, }; cout << "The third element in the array is "; cout << *numbers + 3 << endl;</code>	The result will be the the first element 10 added to 3 that will print 13 and this is not correct. The correct answer: <code>cout << *(numbers + 3) << endl;</code>

5. What is wrong in the following program:

Question	Answer
<pre>int *pint; pint = new int[100]; // Allocate memory cin>>pint; delete pint; // Free memory</pre>	<i>Invalid, the correct answer is:</i> <pre>#include <iostream> using namespace std; void main (){ int *pint; pint = new int[4]; // Allocate memory cin>>*pint; delete pint; // Free memory }</pre>

6. True or false:

a) Any mathematical operation, including multiplication and division, may be used on a **pointer**.

Answer : False

b) To calculate the **amount of memory** used by an array, multiply the number of elements by the number of bytes each element uses.

Answer : True

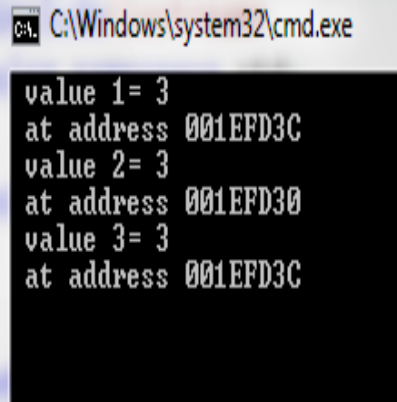
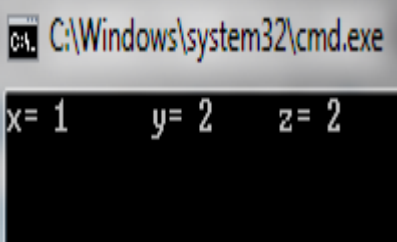
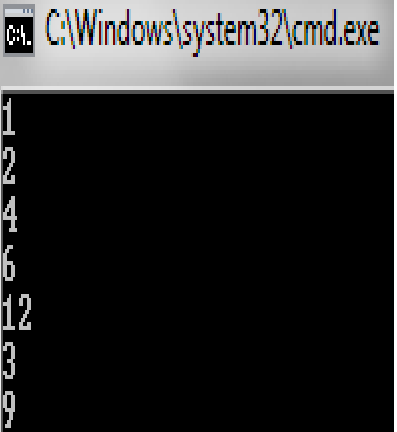
c) A pointer that contains address 0 is called a **null pointer**.

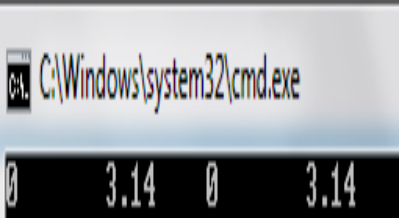
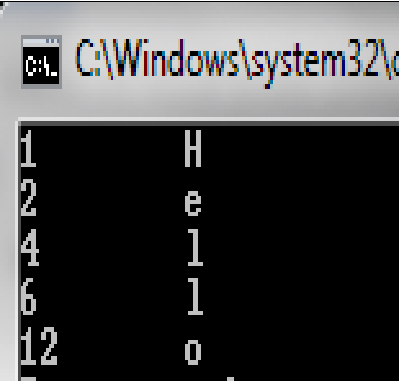
Answer : False

d) When you **add a value to a pointer**, you are actually adding that number to the value stored in the address stored in the pointer.

Answer : False

7. Trace the following codes and write the output for each:

a	<pre>#include<iostream> using namespace std; void main(){ int a=2 , b=3 ; int *z; z = &a ; a = b ; cout<<" value 1= "<< a << "\n at address "<< &a << endl; cout<<" value 2= "<< b << "\n at address "<< &b << endl; cout<<" value 3= "<< *z << "\n at address "<< z << endl;</pre>	
b	<pre>#include <iostream> using namespace std; void duplicate (int a, int *b, int *c){ a=2 ; *b=2 ; *c=2 ; } void main (){ int x=1 , y=3 , z=7 ; duplicate (x, &y, &z); cout<<"x= "<<x<<"\t y= "<<y<<"\t z= "<<z<<endl; }</pre>	
c	<pre>#include <iostream> using namespace std; void main (){ int a[] = {1,2,4,6,12,3,9}; int *z; int i; z = a ; for(i=0 ; i<7; i++){ cout<< *z<<endl ; z++; } }</pre>	

d	<pre>#include <iostream> using namespace std; void main () { float table[] ={1.1,2.3,4.5,6.7,8.7,6.6,4.0,3.3,2.7,4.5 }; float *pt, *qt; pt=table; *pt=0; qt=pt+2; *(pt+2)=3.14; cout<<table[0]<<" \t" << table[2]<< "\t"<< *pt <<"\t"<< *qt<<endl;}</pre>	
e	<pre>#include <iostream> using namespace std; void main (){ int a[] = {1,2,4,6,12}; char c[] = "Hello"; int *z; char *x; int i; z = a; x = c; for(i=0;i<5;i++){ cout<<*z<<"\t"<<*x<<endl ; *z++; x++; }}}</pre>	
f	<pre>#include <iostream> using namespace std; int someFunction(int *m){ *m=(*m)+1; return(*m); } void main (){ int *a,b=1,c=2,d=3,*e; a=&d; e=&c; *a=1; cout<<*e<<"\t"<<c<<"\t"<<d<<endl; b=someFunction(&c); cout<<c<<"\t"<<*e<<endl;}</pre>	